Structural Self-Teaching for Compositional Generation in Unified Multimodal Models

Jaeyoo Park [1] **Bohyung Han**[1,2]

[1]Seoul National University
[2]Samsung Electronics
{bellos1203, bhhan}@snu.ac.kr

Abstract

We propose Structural Self-Teaching (SST), a novel framework that enhances compositional image generation in diffusion-based unified multimodal models (UMMs) by leveraging their internal understanding path as a source of structural supervision. The generation path of UMMs often struggles with dense compositional prompts, leading to missing objects, attribute binding failures, and spatial errors. In contrast, their understanding path localizes the same phrases in images that satisfy these prompts. SST converts this phrase-level grounding into training-time structural signals for the generation path. Specifically, we extract phrase-conditioned support masks from internal attention maps derived from the understanding path and employ them through two mechanisms. Phrase-local contrastive alignment uses these masks as pooling regions and aligns each generated phrase region with the same region of the target image. A structural scaffold builds a phrase-level layout from the same masks and modulates an intermediate spatial state of the generation path during training. The scaffold is phased out during training and removed at inference. SST requires no external structural annotations for training and no additional input at test time. Extensive experiments across multiple UMMs demonstrate that SST improves compositional fidelity over LoRA fine-tuning on the same data, with no inference-time cost. Project page is available at https://bellos1203.github.io/sst

1 Introduction

Unified multimodal models (UMMs) integrate image understanding and generation within a single architecture, and recent systems have shown strong perception performance together with competitive generation quality [1–6]. Despite recent progress, UMMs still struggle to generate images that faithfully satisfy dense compositional prompts. Given prompts that specify multiple objects, attributes, and relations, state-of-the-art UMMs often miscount instances, swap attributes across entities, and drop referenced phrases entirely. These failures intensify as prompts become compositionally denser, motivating training methods that explicitly account for phrase-level structure.

UMMs also provide an internal source of phrase-level structure through their understanding path. Given a training image and its caption, attention between caption and image tokens in the understanding path often localizes individual noun phrases, such as “three flowers,” “a checkered car,” or “a chair,” in spatially coherent image regions (Figure 1), consistent with prior observations in vision-language models [7, 8]. These phrase-conditioned attention maps provide soft spatial supports over visual tokens. Section 3 evaluates whether such supports remain usable under increasing compositional density. We use this internal grounding as a source of structural supervision during generation training, without relying on external annotations.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

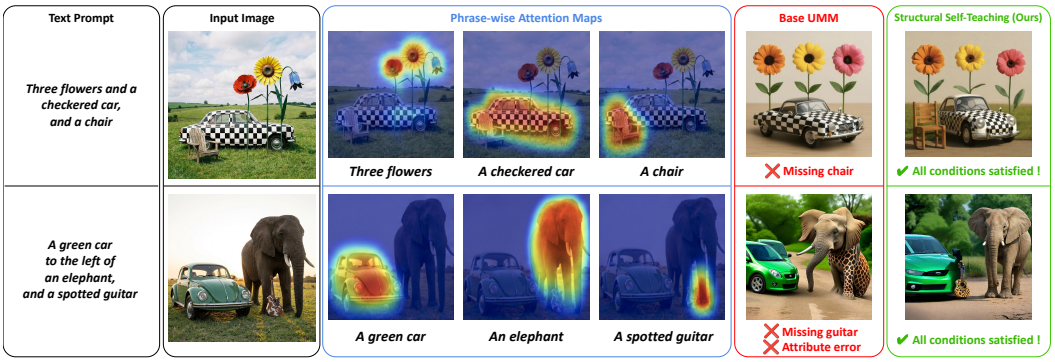


Figure 1: Unified multimodal models can localize phrases they fail to generate. Given an image-text pair that satisfies a compositional prompt, the understanding path produces phrase-wise attention maps that localize the requested concepts. However, generation from the same prompt can omit objects or entangle attributes, as shown by the base UMM. Structural self-teaching transfers these internal supports during training, improving phrase coverage and compositional fidelity.

To expose this internal structure to the generation path, we propose *structural self-teaching* for diffusion-based UMMs, a framework that turns the understanding path into an internal structural teacher for its generation path. The central idea is to extract phrase-level grounding from the understanding path and convert it into structural signals for training the generation pathway. All structural cues are derived from the model itself, requiring neither external structural annotations nor inference-time intervention.

Phrase-level structure reaches the generation path through two routes. The first route is phrase-local contrastive alignment. It uses phrase-conditioned support masks as pooling regions for representation learning. It pulls each phrase-specific region in the generated representation toward the corresponding region in the target representation, while pushing it away from other phrase-specific regions in the same prompt. The second is a structural scaffold, which converts the same support masks into a dense phrase-level canvas and modulates intermediate spatial states during training. Scaffold dropout prevents reliance on the canvas. The scaffold branch is removed at inference. Together, the two mechanisms transfer understanding-side structure through both phrase-level representation alignment and spatial modulation during denoising.

The main contributions of our paper are summarized as follows:

- We propose *structural self-teaching*, a training framework that extracts phrase-conditioned support masks from the understanding path of a UMM and reuses them to supervise its generation path without external structural annotations or test-time inputs.
- We introduce phrase-local contrastive alignment, which leverages the extracted masks to align phrase-specific generated and reference visual representations while preventing semantic collapse among distinct phrase-level features within the same prompt.
- We develop an attenuated structural scaffold that converts these supports into a dense phrase-level canvas for training-time modulation, with a keep probability that decays over training.
- The proposed approach demonstrates consistent improvements in compositional fidelity across multiple UMMs, effectively addressing complex object, attribute, and spatial constraints without modifying the inference-time interface.

The rest of the paper is organized as follows. Section 2 reviews the related work, followed by our motivation in Section 3. The details of our approach are described in Section 4, and the experimental results are presented in Section 5. Finally, we conclude this paper in Section 6.

2 Related Works

2.1 Unified Multimodal Models

Unified multimodal models (UMMs) integrate visual understanding and generation within a single backbone, and generate images either autoregressively over discrete tokens [2–4, 9] or through a diffusion process [1, 10, 5, 6]. Despite competitive performance on both pathways, UMMs exhibit a persistent asymmetry in which they understand more reliably than they generate, prompting a body of post-training work aimed at closing this gap. HermesFlow [11] first quantifies this asymmetry and proposes preference optimization (Pair-DPO) using self-curated preference pairs from both pathways, aligning the two capabilities at the prompt level. SRUM [12] uses the model's own understanding module to derive global and local rewards for reward-weighted generation training. RecA [13] instead conditions the UMM on its own understanding embeddings and reconstructs the input image, providing dense pixel-level supervision without captions. Our method also leverages internal signals from the understanding path, but targets compositional structure rather than reward-based post-training or pixel-level reconstruction: we transfer phrase-conditioned spatial supports through a phrase-local contrastive objective, directly addressing the entanglement and omission failures that emerge as compositional complexity grows.

2.2 Visual Grounding and Attention in MLLMs

Multimodal transformers encode useful structural signals in their attention patterns [14, 7, 15, 8]. In multimodal encoders, word-conditional attention localizes caption words well enough to guide representation learning without region annotations [7]. In frozen LVLMs, Kang et al. [8] show that a small subset of attention heads acts as localization heads and yield strong visual grounding without grounding-specific training, indicating that phrase-conditioned structure already emerges inside the model. At the same time, Visual Attention Sink [16] observes that a few uninformative tokens absorb disproportionate attention mass, meaning raw maps mix signal with systematic artifacts. The practical question is therefore how to extract this structure selectively. We adopt head scoring and sink suppression from previous works [8, 16], and extend them by scoring and selecting heads per phrase rather than globally—since different phrases in the same image are grounded by different heads—and by replacing hard thresholding with soft, temperature-controlled aggregation so that informative sub-threshold maps still contribute to a dense training target.

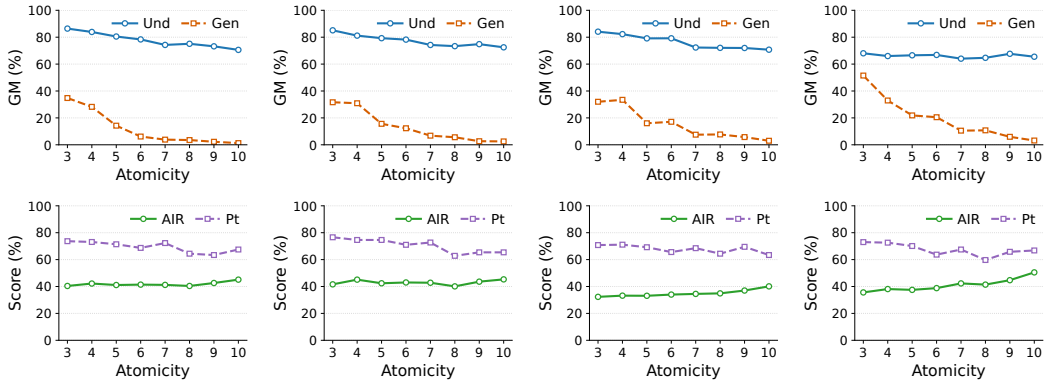
2.3 Compositional Text-to-Image Generation

Even strong generators struggle with compositional prompts, exhibiting missing entities, incorrect attribute binding, unreliable counting, and broken spatial relations—failures that recent benchmarks such as TIFA [17], GenEval [18], and GenEval2 [19] make quantitatively visible. Existing remedies act either during sampling or through added inputs. Prompt-to-Prompt [20] and Attend-and-Excite [21] manipulate cross-attention in each sampling run and leave the trained model unchanged. ControlNet [22] and GLIGEN [23] condition generation on external structure such as edges, depth, or bounding boxes, and LayoutGPT [24] produces such layouts with an LLM planner. These methods require a structural input beyond the prompt, from the user or from a separate planner. Our approach instead regularizes the generation path during training with structure from the model's own understanding path. It adds no inference-time step and no input beyond the prompt.

3 Motivation: Using Understanding Path as Structure Supervision

Our framework uses the understanding path of a UMM as an internal source of phrase-level structural supervision. We therefore first examine whether this path provides usable compositional cues under dense prompts, considering two axes. The first is whether prompt-specified visual facts remain measurable when the corresponding image evidence is available, and the second is whether these facts are spatially grounded enough to define phrase-level supports.

We use the atomic questions from GenEval2 [19], which decompose each prompt into object, attribute, count, and spatial facts. Since GenEval2 provides no oracle images, we curate them through model-based generation and editing to satisfy all atomic conditions (see Appendix C.3 for details). These oracle images serve only as diagnostic inputs for probing the understanding path, and are not used



(a) BLIP-3o 4B (b) BLIP-3o 8B (c) Show-o2 7B (d) BAGEL

Figure 2: Diagnostic of the understanding-path supervision source across four UMM backbones. **Top row.** Prompt-only generation scores (Gen) decrease sharply with atomicity, while understanding-path scores on oracle images (Und) remain comparatively stable. **Bottom row.** Activation-inside ratio (AIR) and center-of-mass pointing accuracy (Pt) of extracted phrase supports against Grounded-SAM pseudo-references remain within narrow bands across atomicity. The top row is not a direct comparison between Und and Gen, see Section 3.

for training or as a generation benchmark. Given each oracle image and prompt, the understanding path answers the corresponding atomic questions, and we separately report prompt-only generation scores under the GenEval2 protocol to indicate the compositional difficulty of the same prompts. We emphasize that Figure 2 serves as a diagnostic of the source signal used by our method, not as a direct comparison between perception and generation.

Across BLIP-3o [6], Show-o2 [5], and BAGEL [1], prompt-only generation scores decrease sharply as atomicity increases, dropping to near-zero on the densest prompts (Figure 2, top row). In contrast, understanding-path scores on the corresponding oracle images remain comparatively stable across the same atomicity range, with only mild degradation at high atomicity. This stability holds across all evaluated backbones, despite their differences in architecture and scale. The understanding path therefore retains measurable compositional information even under high atomicity, supporting its use as an internal source of structural supervision.

We next examine the second axis, whether the source signal is spatially grounded and stable across compositional density. We extract phrase-conditioned support masks from the understanding path (following the procedure detailed in Section 4.1) and compare them against pseudo-reference masks obtained by running Grounded-SAM [25] on the same noun phrases. Across atomicity levels 3–10, the activation-inside ratio and center-of-mass pointing accuracy remain within narrow bands across all four backbones, with no systematic degradation as compositional density grows (Figure 2, bottom row). Since Grounded-SAM pseudo-references are themselves imperfect, we focus on stability under increasing atomicity rather than on absolute localization accuracy. These supports form spatially coherent regions over visual tokens, which we use as pooling regions for phrase-local alignment and as spatial carriers for a structural scaffold during generation training.

4 Structural Self-Teaching for Compositional Generation

The proposed Structural Self-Teaching (SST) framework transfers phrase-level structural cues from the understanding path of a UMM to its generation path during training (Figure 3). Given an image and its corresponding caption, we first extract noun phrases and derive phrase-conditioned support masks from internal text-to-image attention maps within the understanding path (Section 4.1). The support masks reach the generation path through two routes. Phrase-local contrastive alignment uses them as pooling regions for a representation-level target (Section 4.2). The structural scaffold uses them as a spatial canvas that modulates an intermediate generation state during training (Section 4.3). Attenuated scaffold dropout phases the scaffold out during training. The scaffold branch is removed at inference, so the original generation interface requires no additional input. The final objective combines the model's native denoising loss with the phrase-local contrastive loss (Section 4.4).

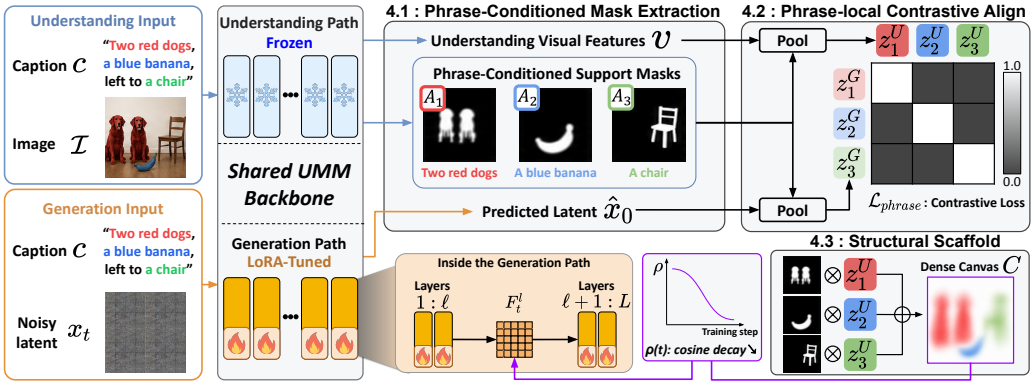


Figure 3: Overview of structural self-teaching. Phrase-conditioned support masks $\{A_p\}$ extracted from the frozen understanding path provide two training-time signals for the LoRA-tuned generation path: phrase-local contrastive alignment between reference and generated phrase representations, and a structural scaffold that modulates an intermediate spatial feature F_t^ℓ. The scaffold is applied with sample-wise cosine-decayed dropout and removed at inference.

4.1 Phrase-Conditioned Support Mask Extraction

To construct a reliable teacher signal from the understanding path, we extract a soft spatial support mask for each noun phrase in the caption. Our extraction procedure builds upon the attention-head scoring criteria of [8], originally proposed for zero-shot visual grounding, but we introduce two critical adaptations for compositional generation. First, we perform candidate scoring and selection per phrase rather than identifying single head set for the entire prompt; this reflects the observation that different phrases within the same complex scene are often grounded by distinct sets of heads. Second, we replace rigid threshold-based selection with a soft, temperature-controlled aggregation, ensuring that informative sub-threshold maps still contribute to the final spatial support.

Given an image $\mathcal{I}$ and its corresponding caption c, we extract noun phrases $\{p_n\}_{n=1}^N$ from c and collect text-to-image attention from every layer-head pair of the understanding path. This produces a candidate pool $A_{p,\ell,h} \in \mathbb{R}^M$ for each phrase p and layer-head pair (ℓ, h), where M denotes the number of image tokens. Following [16], we preprocess each candidate map to suppress background activations and visual attention sinks, then rank these resulting candidate maps within each phrase based on a combined score $s_{p,\ell,h}$ of relevance and spatial plausibility as:

$$s_{p,\ell,h} = s^{\text{sum}}_{p,\ell,h} + s^{\text{ent}}_{p,\ell,h}, \tag{1}$$

where the relevance term $s^{\text{sum}}_{p,\ell,h}$ and the spatial plausibility term $s^{\text{ent}}_{p,\ell,h}$ are both normalized within the phrase-specific candidate pool to ensure comparability.

Let $\bar{A}_{p,k}$ be the k-th ranked candidate map after the aforementioned normalization. The final phrase-conditioned support mask A_p is then constructed by softly aggregating the top-K candidate maps with temperature-controlled weights:

$$A_p = \text{Norm}\left(\sum_{k=1}^{K} w_{p,k}\, \bar{A}_{p,k}\right), \qquad w_{p,k} = \frac{\exp(s_{p,k}/\tau)}{\sum_{k'=1}^{K} \exp(s_{p,k'}/\tau)}, \tag{2}$$

where the temperature $\tau > 0$ controls the sharpness of the aggregation, and $\text{Norm}(\cdot)$ denotes a final sum-to-one renormalization step to ensure the mask values stay within a valid range for the subsequent alignment stage.

The resulting A_p serves as a phrase-conditioned soft spatial support, rather than a binary segmentation or instance mask. By reliably providing coarse localization and separating spatial responsibilities across phrases, A_p acts as a frozen, structurally-grounded teacher that guides the generation path. Implementation details and exact forms of s^{sum} and s^{ent} are provided in Appendix B. Example visualizations are shown in Appendix E.1.

4.2 Phrase-local Contrastive Alignment

We leverage the phrase-conditioned support masks A_p as pooling operators to provide representation-level supervision. Given a caption c, its corresponding ground-truth image I, and a sampled timestep t, the generation path predicts a denoised latent $\hat{x}_0$ from x_t conditioned on c. We encode I into reference visual features $v \in \mathbb{R}^{M \times D}$. For each noun phrase p in the caption, we derive a reference visual representation z_p^U from the target image and a generated visual representation z_p^G from the denoised latent via mask-weighted pooling:

$$z_p^U = \mathrm{Pool}(v, A_p), \qquad z_p^G = g_\phi(\mathrm{Pool}(\hat{x}_0, A_p)), \tag{3}$$

where $\mathrm{Pool}(\cdot, A_p)$ denotes mask-weighted spatial averaging, with A_p resized to match the spatial resolution of the respective feature maps. The projection head g_ϕ maps the pooled features from the generation path into the visual feature space of v. The choice of features and projection is backbone-specific (Appendix A.2).

The alignment objective pulls each generated representation toward its own reference and away from the references of other phrases in the same prompt:

$$\mathcal{L}_{\text{phrase}} = -\frac{1}{N} \sum_{p=1}^{N} \log \frac{\exp(\mathrm{sim}(z_p^G, z_p^U)/\tau_{\text{align}})}{\sum_{q=1}^{N} \exp(\mathrm{sim}(z_p^G, z_q^U)/\tau_{\text{align}})}, \tag{4}$$

where $\mathrm{sim}(\cdot, \cdot)$ denotes cosine similarity and $\tau_{\text{align}} > 0$ is a temperature. The positive term matches the content generated in each phrase region to the corresponding region of the reference image. The in-prompt negatives keep distinct phrases from collapsing onto one representation, which targets the attribute leakage and object fusion observed in dense prompts.

4.3 Attenuated Structural Scaffold

The structural scaffold modulates an intermediate spatial state of the generation path with a phrase-level layout during training. The layout assigns each phrase's reference representation to its support region. Phrase-local contrastive alignment specifies what each phrase region should contain, but it reaches the generation path only through the gradient of a pooled loss. We attenuate the scaffold over training and remove it at inference, so the generation path at inference is unchanged.

Scaffold construction Given the support masks $\{A_p\}_{p=1}^N$ and reference representations $\{z_p^U\}_{p=1}^N$, we first resize each mask to the target resolution and normalize it to $\tilde{A}_p(u) \in [0, 1]$, ensuring the peak activation of each mask is scaled to unity. Each reference representation is projected into a scaffold-specific feature space:

$$e_p = W_s z_p^U, \tag{5}$$

where W_s is a learnable linear projection matrix that aligns the reference representation with the scaffold feature dimension. The scaffold canvas C at spatial location u is then defined as a weighted sum of these features:

$$C(u) = \sum_{p=1}^{N} \tilde{A}_p(u) e_p. \tag{6}$$

By not normalizing the sum across phrases, $C(u)$ naturally retains the support confidence, where regions with stronger phrase support produce higher-magnitude scaffold features.

Scaffold injection The scaffold canvas is injected into an intermediate spatial feature map $F_t^\ell \in \mathbb{R}^{H \times W \times D}$ in the generation path through a lightweight residual adapter:

$$F_t^\ell(u) \leftarrow F_t^\ell(u) + m \cdot \alpha \cdot W_c C(u), \tag{7}$$

where W_c is a projection layer, α is a learnable scalar initialized to zero, and $m \sim \mathrm{Bernoulli}(\rho)$ is a scaffold dropout variable with the keep probability $\rho \in [0, 1]$. The residual form exposes the canvas without changing the architecture of the generation path. Since α is initialized to zero, the injection leaves the forward computation unchanged at the start of training.

We attenuate the injection by decreasing ρ over training with a cosine schedule. Because m is sampled per example and independently of the timestep, scaffolded and unscaffolded steps update the same generation parameters. As ρ decays, unscaffolded steps dominate, so the parameters shaped with the scaffold converge toward the path used at inference.

Table 1: Performance comparison on GenEval2. We report arithmetic mean (AM), geometric mean (GM), category-level scores, and atomicity-wise GM. LoRA FT and SST report means over three training runs, with sample standard deviations for AM and GM. Shaded rows are reference baselines under different settings. CoT modifies inference, and RecA tunes the generation parameters directly, without LoRA, with caption-free reconstruction supervision. Bold marks the best value in each backbone group.

Method	AM	GM	Category					Atomicity							
			Obj.	Attr.	Count	Pos.	Verb	3	4	5	6	7	8	9	10
BLIP-3o 4B	53.87	11.78	77.94	52.87	30.72	35.06	15.44	34.78	28.21	14.22	6.14	3.84	3.53	2.25	1.26
+ LoRA FT	54.70$_{\pm 0.18}$	13.00$_{\pm 0.35}$	76.62	54.24	31.64	36.33	13.37	39.34	28.24	14.65	10.71	4.33	3.46	1.90	1.44
+ SST (Ours)	**58.29**$_{\pm 0.15}$	**16.13**$_{\pm 0.28}$	**81.29**	**54.49**	**37.17**	**41.53**	**17.03**	**47.41**	**33.19**	**16.92**	**14.43**	**5.72**	**4.53**	**3.93**	**2.92**
BLIP-3o 8B	59.35	13.50	85.08	57.79	33.46	47.13	20.34	31.63	30.90	15.59	12.30	6.79	5.61	2.67	2.52
+ LoRA FT	60.48$_{\pm 0.23}$	14.86$_{\pm 0.48}$	86.31	58.00	34.84	46.94	**21.22**	33.05	32.77	16.25	13.87	7.49	**6.92**	**5.23**	3.27
+ SST (Ours)	**61.82**$_{\pm 0.45}$	**16.47**$_{\pm 0.39}$	**87.88**	**58.86**	**35.22**	**50.17**	17.14	**35.78**	**35.92**	**19.84**	**15.83**	**8.63**	6.88	5.14	**3.77**
Show-o2 7B	63.43	15.32	88.36	60.51	44.01	44.05	**27.58**	31.99	33.44	16.01	17.09	7.52	**7.69**	5.74	3.00
+ LoRA FT	63.42$_{\pm 0.38}$	16.03$_{\pm 0.41}$	88.52	61.34	43.91	43.44	18.12	37.44	32.22	18.82	16.66	**8.85**	6.02	5.33	2.88
+ SST (Ours)	**64.28**$_{\pm 0.32}$	**17.11**$_{\pm 0.58}$	**89.44**	**62.71**	**44.12**	**46.27**	20.89	**39.36**	**34.67**	**19.81**	**17.82**	8.45	7.56	**5.86**	**3.31**
BAGEL (14B)	66.36	19.65	89.27	61.90	49.73	44.56	22.96	51.48	32.94	21.85	20.51	10.51	10.75	5.93	3.18
+ CoT	70.81	23.29	91.97	**71.70**	53.07	51.51	17.99	51.94	41.58	30.74	25.55	11.99	10.97	7.45	6.10
+ RecA	72.66	25.52	**94.21**	71.63	55.65	51.55	**27.34**	58.89	44.04	28.71	**30.20**	15.12	10.84	7.77	**8.58**
+ LoRA FT	73.01$_{\pm 0.31}$	26.47$_{\pm 0.21}$	93.23	68.43	57.39	52.71	25.32	59.89	48.71	30.74	28.86	**15.89**	11.50	8.93	7.20
+ SST (Ours)	**74.06**$_{\pm 0.25}$	**27.33**$_{\pm 0.42}$	93.39	71.69	**58.23**	**54.21**	26.82	**60.52**	**52.16**	**31.86**	28.74	14.89	**13.72**	**9.33**	7.44

4.4 Training Objective and Inference

The training objective comprises the phrase-local contrastive loss and the standard denoising loss:

$$\mathcal{L} = \mathcal{L}_{\text{gen}} + \lambda \mathcal{L}_{\text{phrase}}, \tag{8}$$

where $\mathcal{L}_{\text{gen}}$ represents the native denoising objective of the base model and λ is a weighting hyperparameter. During training, the understanding path and the support extraction module remain frozen, while we jointly update the generation path, the projection head g_ϕ, and the scaffold adapter. The scaffold does not introduce an explicit loss term; instead, it modulates the training process solely through the forward computation when active. At inference, both the contrastive objective and the scaffold branch are discarded. The final model thus operates with its original generation interface and requires no additional inputs or computational overhead during deployment. Together, these components define our Structural Self-Teaching (SST) framework, which transfers phrase-level structure from the understanding path to the generation path entirely through internal signals.

5 Experiments

5.1 Setup

We train on the Instruction-Following subset of Echo-4o [26] (70K synthetic image-caption pairs) and evaluate on GenEval2 [19], which contains 800 prompts stratified by atomicity from 3 to 10 and uses Soft-TIFA, an atom-level VQA judge based on Qwen3-VL-8B. GenEval2 measures compositional fidelity, that is, whether the image satisfies each atom specified in the prompt. We report arithmetic mean (AM), geometric mean (GM), category-level scores, and atomicity-stratified GM, using GM as the primary metric since it penalizes missing visual atoms. GenEval++ results are reported in Appendix D.1. We validate four UMM backbones (BLIP-3o 4B/8B, Show-o2 7B, BAGEL) under LoRA-based tuning of the generation modules with the understanding path frozen. Backbone-specific instantiations and training details are provided in Appendix A.

5.2 Quantitative Results

We compare SST with the base UMMs and with LoRA FT, which applies the same LoRA tuning on the same data with the native denoising loss alone (Table 1). For BAGEL, we also report CoT prompting and RecA as reference baselines, since they differ from our setting in inference procedure or tuning protocol.

The gains of SST come from its structural supervision rather than from the training data. LoRA FT trains on the same data, and SST improves over it in AM and GM on all four backbones, which span

Table 2: Results of main ablation studies on the phrase-local contrastive alignment ($\mathcal{L}_{phrase}$) and structural scaffold on BLIP-3o 4B and BAGEL. Means and sample standard deviations over three training runs. Bold marks the best value in each column for each backbone.

	Method	AM	GM	Object	Attribute	Count	Position	Verb
BLIP-3o 4B	LoRA FT	$54.70_{\pm 0.18}$	$13.00_{\pm 0.35}$	$76.62_{\pm 0.42}$	$54.24_{\pm 0.68}$	$31.64_{\pm 0.44}$	$36.33_{\pm 0.59}$	$13.37_{\pm 0.66}$
	Phrase-local only	$56.04_{\pm 0.34}$	$15.42_{\pm 0.41}$	$79.88_{\pm 0.52}$	$54.23_{\pm 0.37}$	$35.77_{\pm 0.66}$	$37.98_{\pm 0.31}$	$\mathbf{17.28}_{\pm 0.84}$
	Scaffold only	$55.24_{\pm 0.28}$	$14.43_{\pm 0.33}$	$77.42_{\pm 0.34}$	$53.94_{\pm 0.24}$	$33.12_{\pm 0.56}$	$38.67_{\pm 0.44}$	$14.42_{\pm 0.83}$
	Full SST	$\mathbf{58.29}_{\pm 0.15}$	$\mathbf{16.13}_{\pm 0.28}$	$\mathbf{81.29}_{\pm 0.25}$	$\mathbf{54.49}_{\pm 0.20}$	$\mathbf{37.17}_{\pm 0.59}$	$\mathbf{41.53}_{\pm 0.42}$	$17.03_{\pm 0.78}$
BAGEL	LoRA FT	$73.01_{\pm 0.31}$	$26.47_{\pm 0.21}$	$93.23_{\pm 0.41}$	$68.43_{\pm 0.49}$	$57.39_{\pm 1.27}$	$52.71_{\pm 0.55}$	$25.32_{\pm 1.09}$
	Phrase-local only	$73.69_{\pm 0.30}$	$27.12_{\pm 0.39}$	$\mathbf{93.47}_{\pm 0.35}$	$70.24_{\pm 0.58}$	$58.20_{\pm 0.41}$	$52.78_{\pm 0.36}$	$26.47_{\pm 0.71}$
	Scaffold only	$73.44_{\pm 0.32}$	$26.79_{\pm 0.37}$	$93.33_{\pm 0.22}$	$69.98_{\pm 0.24}$	$\mathbf{58.42}_{\pm 0.66}$	$53.92_{\pm 0.38}$	$26.14_{\pm 0.62}$
	Full SST	$\mathbf{74.06}_{\pm 0.25}$	$\mathbf{27.33}_{\pm 0.42}$	$93.39_{\pm 0.28}$	$\mathbf{71.69}_{\pm 0.76}$	$58.23_{\pm 0.74}$	$\mathbf{54.21}_{\pm 0.51}$	$\mathbf{26.82}_{\pm 0.67}$

Table 3: Mask source on BLIP-3o 4B (single run each). Grounded-SAM masks serve as an external reference that replaces the attention-derived supports during training, with all other settings fixed. The last row reports the difference between the attention-derived supports and the external reference.

Mask source	AM	GM	Object	Attribute	Count	Position	Verb
None (LoRA FT)	54.56	13.35	76.22	53.60	31.40	36.29	13.60
Grounded-SAM (external)	59.58	16.84	83.09	55.68	35.28	40.52	18.64
Attention-derived (SST)	58.34	16.05	81.43	54.34	37.32	41.57	16.29
Δ (SST $-$ external)	-1.24	-0.79	-1.66	-1.34	$+2.04$	$+1.05$	-2.35

three generation architectures. The gains exceed the standard deviation across three training runs. On BLIP-3o 4B and BAGEL, paired bootstrap intervals over prompts exclude zero (Appendix D.3), and a blinded human evaluation also favors SST (Appendix D.4).

The category-level scores follow what the supports cover. The supports localize noun phrases, and SST scores higher than LoRA FT on Object, Attribute, Count, and Position on all four backbones. Verb, which the supports do not encode and only 82 of the 800 prompts contain, is the exception. On Show-o2, LoRA fine-tuning lowers Verb sharply and SST recovers only part of the drop; on BLIP-3o 8B, SST falls below both baselines. On BAGEL, SST exceeds both reference baselines in AM and GM without changing the inference procedure, while RecA leads on Object and Verb.

5.3 Ablation Studies and Analysis

We analyze the contribution of each component, the source of the support masks, and the design choices of the scaffold, and we examine whether SST reduces generation diversity. Table 2 reports means over three training runs on BLIP-3o 4B and BAGEL. Unless noted otherwise, the remaining ablations use a single run on BLIP-3o 4B, with the same training data, number of optimization steps, and evaluation protocol as the main experiments.

Effect of each component Table 2 ablates Phrase-Local Contrastive Alignment and the Structural Scaffold on BLIP-3o 4B and BAGEL. Phrase-local alignment improves GM over LoRA FT on both backbones, and the full model achieves the highest GM. The two mechanisms overlap in the categories they improve but differ in where their gains concentrate. On BLIP-3o 4B, phrase-local alignment accounts for most of the gain on Object, Count, and Verb. The scaffold contributes most clearly on Position: scaffold only ranks below phrase-local only in GM on both backbones, yet above it on Position. Adding the scaffold to phrase-local alignment raises Position beyond run-to-run variation on both backbones, and on BLIP-3o 4B it also raises Object and Count. On BAGEL, phrase-local alignment alone leaves Position at the level of LoRA FT.

Support masks Table 3 replaces the attention-derived supports with masks from an external segmentation model, Grounded-SAM, keeping all other settings fixed. The attention-derived supports come within 0.79 GM of the external masks and achieve 77% of their gain over LoRA FT. Coarse supports from the understanding path suffice for SST, without an external segmentation model.

Table 4: Ablation on λ, the weighting of $\mathcal{L}_{\text{phrase}}$. The optimal scale of λ depends on the backbone. We report Overall GM (%) on GenEval2 and bold the best λ per backbone (single run each).

BLIP-3o variants					Show-o2 / BAGEL				
λ	1	5	10	15	λ	0.01	0.1	0.2	0.5
BLIP-3o 4B	14.11	15.36	**16.05**	14.71	Show-o2 7B	15.66	**16.55**	16.12	15.44
BLIP-3o 8B	14.55	16.18	**16.89**	15.23	BAGEL	26.14	**27.21**	26.40	25.76

Table 5: Scaffold design ablations on GenEval2. We evaluate scaffold injection layer and dropout schedule while keeping all other settings fixed. Since the scaffold modulates spatial generation states, we report Position and Overall GM. Bold numbers indicate the best performance in each column.

(a) Scaffold injection layer

Layer	Position	Overall (GM)
No scaffold	38.27	15.48
0	**41.57**	**16.05**
3	40.79	15.86
7	40.05	15.45
11	39.46	15.52

(b) Scaffold dropout schedule

Schedule	Position	Overall (GM)
No scaffold	38.27	15.48
Fixed $\rho = 1.0$	38.14	14.28
Fixed $\rho = 0.5$	40.15	15.76
Fixed $\rho = 0.25$	39.24	15.32
Decay of ρ	**41.57**	**16.05**

Loss weighting We ablate λ, the weighting of $\mathcal{L}_{\text{phrase}}$. The optimal scale differs across backbones, with BLIP-3o variants preferring $\lambda \sim 10$ and Show-o2 and BAGEL preferring $\lambda \sim 0.1$ (Table 4). The split follows the generation target space: BLIP-3o denoises in the EVA-CLIP feature space, whereas Show-o2 and BAGEL predict velocities in a VAE latent space. We use the best λ per backbone in all main experiments.

Scaffold injection layer Table 5(a) evaluates different injection layers for the structural scaffold. Since the scaffold modulates spatial generation states, we report Position as the most directly relevant category together with Overall GM. The earliest layer performs best, and Position decreases as the scaffold is moved to later layers, suggesting that scaffold modulation is most effective before the generation path has committed to a spatial layout within each forward pass.

Scaffold scheduling Table 5(b) compares scaffold dropout schedules. Since the scaffold branch is removed at inference, the schedule determines how much the generation path is exposed to the privileged scaffold signal during training. The decay schedule achieves the best Position and Overall GM, outperforming fixed keep probabilities. This suggests that the scaffold should provide stronger guidance early in training and be gradually phased out as the unscaffolded generation path is optimized. The result supports scaffold dropout as a training-time curriculum rather than a persistent conditioning pathway.

Scaffold mechanism We isolate the source of the scaffold benefit through canvas controls. Replacing the phrase-grounded canvas with a Uniform canvas, which removes spatial structure, yields performance comparable to no-scaffold, while a Shuffled canvas, which preserves spatial structure but randomizes phrase–region binding, falls below it. The full canvas outperforms both controls (Table 6), indicating that the benefit derives from correctly binding phrase content to spatial support, rather than from added conditioning capacity alone.

Table 6: Canvas controls. Uniform removes structure; Shuffled randomizes binding.

Canvas	GM
No scaffold	15.48
Uniform	15.37
Shuffled	14.69
Full (Ours)	**16.05**

Generation diversity Table 7 measures generation diversity among four samples per GenEval2 prompt with two distances, LPIPS [27] and the cosine distance between DINOv2 [28] features. Diversity depends mainly on the backbone, where BAGEL yields roughly twice the distance of BLIP-3o 4B under both metrics. Within each backbone, SST yields higher diversity than LoRA FT under both metrics on BLIP-3o 4B. On BAGEL, LPIPS is similar between the two methods, while DINOv2 distance is higher with SST.

Table 7: Generation diversity on GenEval2 (800 prompts). For each prompt, we generate four samples and average the distance over all six pairs: LPIPS, and the cosine distance between DINOv2 features. Higher is more diverse. Subscripts are 95% bootstrap intervals over prompts.

	BLIP-3o 4B		BAGEL	
Method	LPIPS	DINOv2	LPIPS	DINOv2
Base	$0.321_{[0.316, 0.326]}$	$0.127_{[0.121, 0.133]}$	$0.586_{[0.579, 0.592]}$	$0.300_{[0.293, 0.308]}$
LoRA FT	$0.283_{[0.275, 0.291]}$	$0.098_{[0.093, 0.102]}$	$0.628_{[0.618, 0.638]}$	$0.243_{[0.235, 0.251]}$
SST (Ours)	$0.317_{[0.308, 0.326]}$	$0.123_{[0.117, 0.128]}$	$0.626_{[0.614, 0.638]}$	$0.256_{[0.249, 0.263]}$

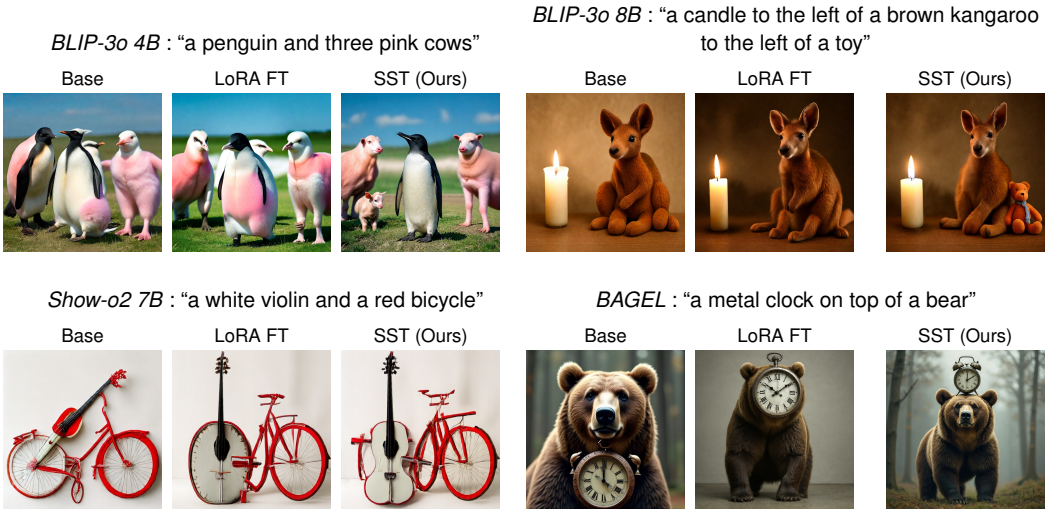


Figure 4: Qualitative comparison across four UMM backbones. Each group of three images shows one failure mode of the base model: attribute mis-binding for BLIP-3o 4B (cows fused into penguin-like hybrids), object omission for BLIP-3o 8B (the toy is missing), object entanglement for Show-o2 7B (violin and bicycle fused; LoRA FT also fails to fully separate them), and spatial violation for BAGEL (the clock is held rather than placed on top of the bear). SST restores the prompt-specified compositional structure across all cases, while LoRA fine-tuning provides only partial mitigation.

5.4 Qualitative Results

Figure 4 presents qualitative comparisons across the four UMM backbones, illustrating how SST resolves distinct compositional failure modes of the base UMMs. The selected cases span attribute mis-binding, object omission, object entanglement, and spatial violation, covering the major error types observed in dense compositional prompts. Across all cases, the base UMMs either drop, fuse, or mis-relate the prompt-specified concepts, and LoRA fine-tuning provides only partial mitigation while preserving similar failure structures. SST instead restores the prompt-specified compositional structure, generating distinct objects with correct attributes and spatial relations. These cases illustrate the failure modes that SST reduces: omission, fusion, and misplacement of prompt-specified concepts.

6 Conclusion

We present Structural Self-Teaching (SST), a framework that improves compositional generation in diffusion-based unified multimodal models. SST leverages the model's own understanding path as an internal source of phrase-level structural supervision, converting phrase-conditioned attention into support masks without requiring external annotations. The masks reach the generation path through two complementary mechanisms. Phrase-local contrastive alignment aligns each generated phrase region with its target region and separates distinct phrases within a prompt. A training-only structural scaffold modulates an intermediate spatial state with a phrase-level layout built from the same masks. Experiments across representative UMMs demonstrate the effectiveness of SST, improving compositional fidelity across object, attribute, count, and spatial constraints.

Acknowledgement

This work was partly supported by Samsung Electronics Co., Ltd. (IO250418-12669-01), and the Institute of Information & communications Technology Planning & Evaluation (IITP) grants [No.RS-2022-II220959 (No.2022-0-00959), (Part 2) Few-Shot Learning of Causal Inference in Vision and Language for Decision Making, No.RS-2021-II211343, Artificial Intelligence Graduate School Program (Seoul National University), RS-2025-25442338, AI star Fellowship Support Program (Seoul National University)], and the National Research Foundation of Korea (NRF) grant [No.RS-2022-NR070855, Trustworthy Artificial Intelligence], funded by the Korean government (MSIT).

References

[1] Deng, C., Zhu, D., Li, K., Gou, C., Li, F., Wang, Z., Zhong, S., Yu, W., Nie, X., Song, Z., et al.: Emerging properties in unified multimodal pretraining. arXiv preprint arXiv:2505.14683 (2025)

[2] Wu, C., Chen, X., Wu, Z., Ma, Y., Liu, X., Pan, Z., Liu, W., Xie, Z., Yu, X., Ruan, C., Luo, P.: Janus: Decoupling visual encoding for unified multimodal understanding and generation. In CVPR. (2025)

[3] Chen, X., Wu, Z., Liu, X., Pan, Z., Liu, W., Xie, Z., Yu, X., Ruan, C.: Janus-pro: Unified multimodal understanding and generation with data and model scaling. arXiv preprint arXiv:2501.17811 (2025)

[4] Xie, J., Mao, W., Bai, Z., Zhang, D.J., Wang, W., Lin, K.Q., Gu, Y., Chen, Z., Yang, Z., Shou, M.Z.: Show-o: One single transformer to unify multimodal understanding and generation. In ICLR. (2025)

[5] Xie, J., Yang, Z., Shou, M.Z.: Show-o2: Improved native unified multimodal models. In NeurIPS. (2025)

[6] Chen, J., Xu, Z., Pan, X., Hu, Y., Qin, C., Goldstein, T., Huang, L., Zhou, T., Xie, S., Savarese, S., et al.: Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. arXiv preprint arXiv:2505.09568 (2025)

[7] Park, J., Han, B.: Multi-modal representation learning with text-driven soft masks. In CVPR. (2023)

[8] Kang, S., Kim, J., Kim, J., Hwang, S.J.: Your large vision-language model only needs a few attention heads for visual grounding. In CVPR. (2025)

[9] Team, C.: Chameleon: Mixed-modal early-fusion foundation models. arXiv preprint arXiv:2405.09818 (2024)

[10] Zhou, C., Yu, L., Babu, A., Tirumala, K., Yasunaga, M., Shamis, L., Kahn, J., Ma, X., Zettlemoyer, L., Levy, O.: Transfusion: Predict the next token and diffuse images with one multi-modal model. In ICLR. (2025)

[11] Yang, L., Zhang, X., Tian, Y., Shang, C., Xu, M., Zhang, W., Cui, B.: Hermesflow: Seamlessly closing the gap in multimodal understanding and generation. In NeurIPS. (2025)

[12] Jin, W., Niu, Y., Liao, J., Duan, C., Li, A., Gao, S., Liu, X.: Srum: Fine-grained self-rewarding for unified multimodal models. In ECCV. (2026)

[13] Xie, J., Darrell, T., Zettlemoyer, L., Wang, X.: Reconstruction alignment improves unified multimodal models. In ICLR. (2026)

[14] Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., Hoi, S.C.H.: Align before fuse: Vision and language representation learning with momentum distillation. In NeurIPS. (2021)

[15] Park, J., Choi, J.Y., Park, J., Han, B.: Hierarchical visual feature aggregation for ocr-free document understanding. In NeurIPS. (2024)

[16] Kang, S., Kim, J., Kim, J., Hwang, S.J.: See what you are told: Visual attention sink in large multimodal models. In ICLR. (2025)

[17] Hu, Y., Liu, B., Kasai, J., Wang, Y., Ostendorf, M., Krishna, R., Smith, N.A.: TIFA: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In ICCV. (2023)

[18] Ghosh, D., Hajishirzi, H., Schmidt, L.: GenEval: An object-focused framework for evaluating text-to-image alignment. In NeurIPS. (2023)

[19] Kamath, A., Chang, K.W., Krishna, R., Zettlemoyer, L., Hu, Y., Ghazvininejad, M.: Geneval 2: Addressing benchmark drift in text-to-image evaluation. arXiv preprint arXiv:2512.16853 (2025)

[20] Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control. In ICLR. (2023)

[21] Chefer, H., Alaluf, Y., Vinker, Y., Wolf, L., Cohen-Or, D.: Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. In SIGGRAPH. (2023)

[22] Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In ICCV. (2023)

[23] Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., Li, C., Lee, Y.J.: Gligen: Open-set grounded text-to-image generation. In CVPR. (2023)

[24] Feng, W., Zhu, W., Fu, T.J., Jampani, V., Akula, A., He, X., Basu, S., Wang, X.E., Wang, W.Y.: Layoutgpt: Compositional visual planning and generation with large language models. In NeurIPS. (2023)

[25] Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., Zeng, Z., Zhang, H., Li, F., Yang, J., Li, H., Jiang, Q., Zhang, L.: Grounded sam: Assembling open-world models for diverse visual tasks. arXiv preprint arXiv:2401.14159 (2024)

[26] Ye, J., Jiang, D., Wang, Z., Zhu, L., Hu, Z., Huang, Z., He, J., Yan, Z., Yu, J., Li, H., et al.: Echo-4o: Harnessing the power of gpt-4o synthetic images for improved image generation. arXiv preprint arXiv:2508.09987 (2025)

[27] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In CVPR. (2018)

[28] Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: DINOv2: Learning robust visual features without supervision. TMLR (2024)

[29] Sun, Q., Fang, Y., Wu, L., Wang, X., Cao, Y.: Eva-clip: Improved training techniques for clip at scale. arXiv preprint arXiv:2303.15389 (2023)

[30] Zhuo, L., Du, R., Xiao, H., Li, Y., Liu, D., Huang, R., Liu, W., Zhao, L., Wang, F.Y., Ma, Z., et al.: Lumina-next: Making lumina-t2x stronger and faster with next-dit. In NeurIPS. (2024)

[31] Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training. In ICCV. (2023)

[32] Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)

Table 8: Training configuration across backbones.

Backbone	LoRA r	LoRA α	Learning rate
BLIP-3o 4B	64	16	1×10^{-4}
BLIP-3o 8B	64	16	5×10^{-5}
Show-o2 7B	32	16	1×10^{-5}
BAGEL	64	16	1×10^{-4}

A Implementation Details

A.1 Training Configuration

Table 8 summarizes the training configuration. All backbones are trained for 5,000 steps with a per-device batch size of 1 across 4 GPUs and 8 gradient accumulation steps, yielding an effective batch size of 32. We use the AdamW optimizer with 200 warmup steps. LoRA rank, alpha, and learning rate vary across backbones as listed below; all other hyperparameters are shared.

A.2 Architecture-Specific Instantiations

We instantiate SST on three unified multimodal models: BLIP-3o, Show-o2, and BAGEL. Each backbone extracts its own support masks $\{A_p\}$ offline from its understanding path, following the same procedure (Appendix B). The canvas construction and the zero-initialized residual adapter of Section 4.3 are also identical across backbones.

The backbones differ in three choices, each of which follows from the native generation pathway. The student feature is the generation-side state that we pool for z_p^G. The reference feature v is the feature of the target image that we pool for z_p^U. The generation module is the submodule that we tune with LoRA and whose first layer receives the scaffold. Table 9 summarizes these choices. In every case, we L2-normalize the pooled representations and apply the objective of Equation 4.

BLIP-3o (cascaded DiT) BLIP-3o's generation path operates natively in the EVA-CLIP [29] feature space, and the DiT directly outputs $\hat{x}_0$ in that space. We attach a projection head g_ϕ that maps $\hat{x}_0$ into the EVA-CLIP prototype space used for contrastive alignment. The DiT cross-attention is tuned with LoRA. The scaffold is injected as a residual on the output of the first NextDiT [30] block, whose token grid is already aligned with the latent spatial layout.

Show-o2 (LLM with diffusion head) Show-o2's diffusion head predicts velocities in a VAE latent space. We recover $\hat{x}_0$ from the predicted velocity, patchify it, and attach a linear projection head g_ϕ that maps each patch into the space of the reference feature, Show-o2's unified visual representation of the clean latent of the target image. The diffusion head is tuned with LoRA and g_ϕ is trained from scratch. The scaffold is injected on the first `diffusion_head_a` block. Because Show-o2 packs `[text | spatial | ...]` in a single sequence, the injection hook slices the spatial sub-region using the recorded modality positions, applies the adapter, and writes the modified state back into the corresponding positions.

BAGEL (LLM-as-DiT) BAGEL has no separate DiT module; the LLM itself predicts velocities for the image latents through a generation-specific MoE branch within each LLM layer. We attach a projection head g_ϕ that maps LLM hidden states to the SigLIP [31] space, producing the student feature for contrastive alignment. We tune the generation MoE of every LLM layer with LoRA, and train g_ϕ from scratch. The scaffold is injected on the first LLM layer. Since BAGEL packs `[text | latent | ...]` in a single sequence, the injection hook acts only on the latent-token positions and leaves text positions untouched.

A.3 Compute Resources

We train BLIP-3o 4B on 4×NVIDIA A5000 GPUs, requiring approximately 11 hours of training. The remaining backbones (BLIP-3o 8B, Show-o2 7B, and BAGEL) are trained on 4×NVIDIA A6000 GPUs, each requiring approximately 10 hours.

Table 9: Backbone-specific instantiation of phrase-local contrastive alignment and structural scaffold injection. All other components, including the support masks, canvas construction, and residual adapter, are shared across backbones.

	BLIP-3o	Show-o2	BAGEL
Student feature	$\hat{x}_0$ (EVA-CLIP)	patchified $\hat{x}_0$ (VAE latent)	final LLM hidden state
Reference feature	EVA-CLIP feature	unified visual representation	SigLIP feature
Trainable modules (LoRA)	DiT cross-attention	diffusion head	generation MoE (all LLM layers)
Scaffold injection point	first NextDiT block	first `diffusion_head_a` block	first LLM layer (latent positions)

Support extraction runs once per backbone and dataset. On BLIP-3o 4B, it takes about 10 hours on 4 A5000 GPUs for the 70K pairs and produces 4.2 GB of cached supports, which all training runs reuse. Extraction materializes attention maps, which precludes fused attention kernels such as FlashAttention. During training, the frozen encoder computes the reference features without gradients, and on BLIP-3o the denoising target already provides them.

B Phrase-Conditioned Support Mask Extraction

This appendix provides the full specification of the mask extraction pipeline summarized in Section 4.1. The scoring criteria follow [8], while the per-phrase application and soft aggregation are introduced in this work.

For each training pair $(\mathcal{I}, c)$, we extract noun phrases $\{p_n\}_{n=1}^{N}$ from c using spaCy, and collect text-to-image attention from every layer-head pair of the understanding path. We denote the resulting candidate map for phrase p at layer ℓ and head h as $A_{p,\ell,h} \in \mathbb{R}^M$, where M is the number of image tokens. Candidate scoring and selection are performed independently for each phrase, since different phrases in the same image are typically grounded by different heads.

B.1 Attention Map Preprocessing

Each candidate map is preprocessed in two steps before scoring. Both steps are adopted from prior work and used as filtering only.

Mean-centered rectification We suppress weak background activations by subtracting the per-map mean and applying a non-negative rectification:

$$\tilde{A}_{p,\ell,h}(i) = \mathrm{ReLU}\left(A_{p,\ell,h}(i) - \frac{1}{M} \sum_{i'=1}^{M} A_{p,\ell,h}(i') \right). \tag{9}$$

This step retains only above-average activations within each candidate map.

Visual sink removal Following [16], we account for visual attention sinks: a small set of image tokens that consistently absorb large attention mass irrespective of semantic content. We compute a global sink score $g(i)$ for each image token i by averaging $\tilde{A}_{p,\ell,h}(i)$ across the entire candidate pool, and define the sink set $\mathcal{S}$ as the top q fraction of tokens with the largest $g(i)$. For each candidate, we measure the non-sink mass ratio

$$r_{p,\ell,h} = \frac{\sum_{i \notin \mathcal{S}} \tilde{A}_{p,\ell,h}(i)}{\sum_{i=1}^{M} \tilde{A}_{p,\ell,h}(i) + \varepsilon}. \tag{10}$$

Maps with $r_{p,\ell,h}$ below a threshold η are discarded as sink-dominated. The remaining maps are sink-zeroed:

$$\bar{A}_{p,\ell,h}(i) = \begin{cases} 0 & i \in \mathcal{S}, \\ \tilde{A}_{p,\ell,h}(i) & i \notin \mathcal{S}. \end{cases} \tag{11}$$

The preprocessed maps $\{\bar{A}_{p,\ell,h}\}$ form the surviving candidate pool $\mathcal{C}_p$ for each phrase p.

B.2 Per-Phrase Scoring

After preprocessing, each surviving candidate is scored on two criteria: relevance and spatial plausibility. Both scores are computed per phrase and standardized within the corresponding candidate pool $\mathcal{C}_p$, so that scores remain comparable across phrases.

Attention-sum score The attention-sum measures how much mass the candidate places on image tokens for the given phrase. We compute the raw sum

$$u_{p,\ell,h} = \sum_{i=1}^{M} \bar{A}_{p,\ell,h}(i), \tag{12}$$

and standardize within $\mathcal{C}_p$:

$$s^{\text{sum}}_{p,\ell,h} = \frac{u_{p,\ell,h} - \mu^{\text{sum}}_p}{\sigma^{\text{sum}}_p + \varepsilon}, \tag{13}$$

where μ^{sum}_p and σ^{sum}_p are the mean and standard deviation of $\{u_{p,\ell',h'}\}_{(\ell',h')\in\mathcal{C}_p}$.

Entropy score The entropy term penalizes overly diffuse maps. We compute the spatial entropy of the map normalized to unit mass, so that the score depends on spread and not on mass:

$$P_{p,\ell,h}(i) = \frac{\bar{A}_{p,\ell,h}(i)}{u_{p,\ell,h} + \varepsilon}, \qquad H_{p,\ell,h} = -\sum_{i=1}^{M} P_{p,\ell,h}(i) \log\big(P_{p,\ell,h}(i) + \varepsilon\big) \tag{14}$$

and standardize the negative entropy within the same candidate pool:

$$s^{\text{ent}}_{p,\ell,h} = \frac{-H_{p,\ell,h} - \mu^{\text{ent}}_p}{\sigma^{\text{ent}}_p + \varepsilon}, \tag{15}$$

with μ^{ent}_p and σ^{ent}_p defined analogously over $\{-H_{p,\ell',h'}\}$. This formulation favors candidates that are concentrated relative to other candidates for the same phrase, without imposing an absolute compactness assumption.

Final score We sum the two standardized terms with equal weighting:

$$s_{p,\ell,h} = s^{\text{sum}}_{p,\ell,h} + s^{\text{ent}}_{p,\ell,h}. \tag{16}$$

Once candidate scoring is performed per phrase, the two standardized terms operate on comparable scales and play complementary roles, with the sum term capturing relevance and the entropy term penalizing diffuse maps.

B.3 Soft Top-K Aggregation

For each phrase p, we rank candidates by $s_{p,\ell,h}$ and retain the top K maps $\{\bar{A}_{p,k}\}_{k=1}^{K}$ with corresponding scores $\{s_{p,k}\}_{k=1}^{K}$. The phrase-conditioned support mask is obtained by temperature-controlled softmax aggregation:

$$A_p = \sum_{k=1}^{K} w_{p,k}\, \bar{A}_{p,k}, \qquad w_{p,k} = \frac{\exp(s_{p,k}/\tau)}{\sum_{k'=1}^{K} \exp(s_{p,k'}/\tau)}. \tag{17}$$

The aggregated mask is renormalized so that $\sum_{i=1}^{M} A_p(i) = 1$. Compared to hard top-$K$ averaging, soft aggregation preserves informative sub-maximal candidates and produces a denser support, which is particularly useful in the per-phrase regime where the set of useful heads varies across phrases.

B.4 Hyperparameter Values

These values are fixed across all backbones in our experiments. Per-phrase candidate selection allows the pipeline to use a single set of hyperparameters without backbone-specific tuning.

Table 10: Hyperparameters for mask extraction.

Hyperparameter	Symbol	Value
Sink fraction	q	0.05
Non-sink mass threshold	η	0.7
Top-K candidates	K	8
Aggregation temperature	τ	0.07
Numerical stabilizer	ε	10^{-8}

B.5 Output Interpretation

The extracted A_p should be understood as a phrase-conditioned soft spatial support, not as a hard segmentation or an exact instance decomposition. The masks are reliable at coarse localization and at separating spatial responsibility across phrases, and less suitable for recovering finer structural properties such as exact cardinality. We treat A_p as a frozen teacher signal derived from the understanding path, and use it to supervise the generation path through phrase-local contrastive alignment (Section 4.2) and structural scaffolding (Section 4.3).

C Dataset and Evaluation Protocol

C.1 Training Dataset

We use the Instruction-Following subset of Echo-4o [26], a synthetic dataset generated by GPT-4o that contains approximately 70K image-caption pairs. The captions are constructed through template-driven composition over curated base concepts, covering compositional instructions involving multiple objects, attributes (color, material, pattern), counts, sizes, and spatial relations. The synthetic and template-controlled nature of Echo-4o provides clean, diverse compositional supervision that complements the underrepresentation of dense compositional prompts in real-world image-caption corpora. We use the dataset under its publicly released license and apply no additional preprocessing beyond noun phrase extraction with spaCy as described in Appendix B.

C.2 GenEval2 Protocol

GenEval2 [19] contains 800 prompts stratified by atomicity from 3 to 10, with 100 prompts per level. Atomicity counts the number of visual atoms (objects, attributes, relations) specified in a prompt. Each prompt is decomposed into atomic VQA questions covering five skill categories (Object, Attribute, Count, Position, Verb), and generated images are evaluated by Soft-TIFA, an atom-level VQA judge based on Qwen3-VL-8B [32]. We follow the standard protocol and generate one image per prompt with default sampling hyperparameters of each backbone. We report arithmetic mean (AM), geometric mean (GM), category-level scores, and atomicity-stratified GM as defined in the original benchmark.

C.3 Oracle Image Set Curation for Section 3

The understanding-path diagnostic in Section 3 requires images that satisfy every atomic condition of a given prompt. GenEval2 [19] provides prompts and atomic questions but does not release paired images, so we construct an oracle image set ourselves. The set is used solely as input to the understanding path during the diagnostic in Section 3, and is not used to train any of the compared models or to evaluate generation quality.

Prompts We use the GenEval2 prompts unchanged. The set spans atomicity levels from 3 to 10 with 100 prompts at each level, for a total of 800 prompts. Each prompt is paired with a single oracle image, yielding a final oracle set of 800 image-prompt pairs.

Candidate generation For each prompt, we generate candidate images using two strong image generation models, GPT-Image-1 and Nano Banana. Using two independent generators reduces bias toward the stylistic and compositional preferences of a single model, and increases the likelihood that at least one candidate is close to satisfying the prompt before any editing is applied.

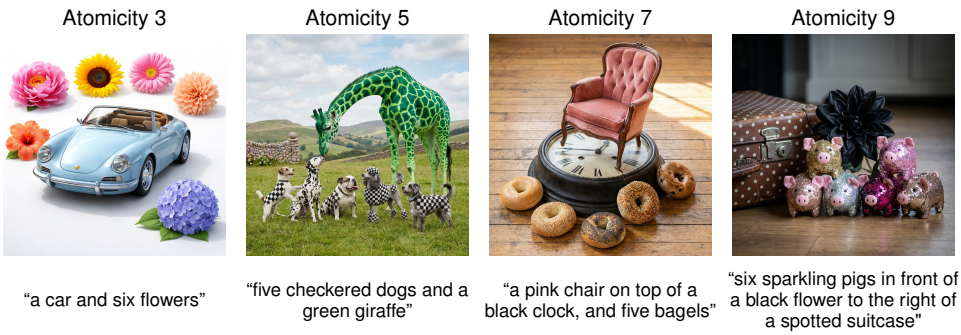


Figure 5: Representative oracle images across atomicity levels. Each image satisfies all atomic conditions of its paired GenEval2 prompt under both the MLLM judge and the authors' visual inspection.

Atomic condition verification Each candidate is evaluated by the same oracle MLLM judge used in the GenEval2 protocol, Qwen3-VL-8B [32], against all atomic questions of the corresponding prompt. Candidates that satisfy every atomic condition are retained as oracle images. Candidates that fail one or more conditions enter the editing stage.

Model-based editing We edit partially satisfying candidates using the same two models (GPT-Image-1 and Nano Banana) in their image-editing modes. Edits are conditioned on a description of the failing atomic condition. After each edit, the result is re-evaluated by the MLLM judge, and the editing loop terminates when all atomic conditions are satisfied or after a small number of unsuccessful iterations. Images that cannot be brought to full satisfaction are discarded rather than retained with partial coverage. No human editing is performed at any stage.

Visual inspection by the authors The authors visually inspect every retained image in the final oracle set. The role of inspection is to catch failure modes that the MLLM judge may miss, such as subtle attribute leakage, unnatural artifacts introduced by editing, or duplicated objects that nominally satisfy a count condition. Images that pass MLLM verification but fail authorial inspection are returned to the editing stage or replaced with a freshly generated candidate. No external workers are involved in either editing or inspection.

Final set statistics The final oracle set contains 800 images, distributed evenly across atomicity levels with 100 images per level. Each retained image satisfies all atomic conditions of its paired prompt under both the MLLM judge and the authors' visual inspection. Figure 5 shows representative examples across atomicity levels.

Scope and limitations. The oracle set is a diagnostic instrument rather than a benchmark. It is constructed at a scale appropriate for probing understanding-path behavior across atomicity levels, not for training or for benchmark-grade evaluation. Both the MLLM judge and the editing models are fallible, and a small number of retained images may exhibit subtle inconsistencies that escape both verification stages. We accept this as the same noise floor that affects the GenEval2 generation evaluation, and the diagnostic in Section 3 should be interpreted in this context. The oracle set is not used as training data or as a generation benchmark anywhere in the paper.

D Additional Quantitative Results

D.1 GenEval++

GenEval++ [26] contains 280 prompts spanning seven task types of increasing difficulty, evaluated by a GPT-4.1-based judge that marks an image correct only if all compositional constraints are satisfied. Table 11 reports per-task and overall accuracy across four UMM backbones. SST improves overall accuracy over both the base UMM and LoRA fine-tuning on all four backbones. The ranking Base $<$ LoRA FT $<$ SST holds across all backbones, indicating that the improvements observed on GenEval2 transfer consistently to this independent compositional benchmark.

Table 11: Performance comparison on GenEval++ [26]. We report per-task accuracy (%) and overall accuracy. Task abbreviations: CA (color attribute), SCA (spatial-count attribute), CSA (color-spatial attribute), CC (color-count), MC (multi-count), SS (size/spatial), Ct (counting). Bold indicates the best result per backbone group.

Method	CA	SCA	CSA	CC	MC	SS	Ct	Overall
BLIP-3o 4B								
Base	12.50	17.50	40.00	15.00	15.00	52.50	27.50	25.71
+ LoRA FT	12.50	20.00	**42.50**	15.00	12.50	55.00	27.50	26.43
+ SST (Ours)	**20.00**	**25.00**	40.00	**17.50**	**17.50**	**62.50**	**32.50**	**30.71**
BLIP-3o 8B								
Base	15.00	**22.50**	45.00	10.00	25.00	62.50	**35.00**	30.71
+ LoRA FT	17.50	20.00	47.50	12.50	30.00	60.00	**35.00**	31.79
+ SST (Ours)	**17.50**	**22.50**	**50.00**	**15.00**	**30.00**	**67.50**	32.50	**33.57**
Show-o2 7B								
Base	27.50	**7.50**	17.50	12.50	12.50	30.00	10.00	16.79
+ LoRA FT	30.00	5.00	17.50	15.00	17.50	30.00	12.50	18.21
+ SST (Ours)	**32.50**	5.00	**20.00**	**15.00**	**17.50**	**35.00**	**12.50**	**19.64**
BAGEL								
Base	45.00	7.50	37.50	27.50	**52.50**	50.00	47.50	38.21
+ LoRA FT	55.00	17.50	**50.00**	37.50	42.50	**55.00**	**57.50**	45.00
+ SST (Ours)	**60.00**	**20.00**	**50.00**	**40.00**	47.50	**55.00**	55.00	**46.79**

Table 12: Mask quality ablation on a 1K Echo-4o subset (BLIP-3o 4B). We progressively add per-phrase head selection and soft top-K aggregation to the global head selection baseline of [8]. Sink suppression [16] is applied to all variants. Bold indicates the best result in each column.

Per-phrase selection	Soft scoring	AIR	Pt
		0.321	0.603
✓		0.371	0.645
✓	✓	**0.411**	**0.718**

D.2 Ablation Studies for Support Mask

Support Mask Quality. We verify mask quality on a uniformly sampled 1,000 subset of Echo-4o, using Grounded-SAM [25] on the same noun phrases as pseudo-references. We report activation-inside ratio (AIR) and center-of-mass pointing (Pt), which align with the use case of our supports as soft pooling regions, and focus on relative improvements since the pseudo-references are themselves imperfect. As shown in Table 12, both per-phrase head selection and soft top-K aggregation provide consistent improvements in AIR and pointing accuracy over global head selection [8].

Sensitivity to extraction hyperparameters. We retrained BLIP-3o 4B with supports extracted under different values of τ and K (Table 13). Aggregation must draw on more than one candidate map. Setting $K = 1$, or $\tau = 0.01$, which concentrates the softmax on the top-ranked map, lowers GM by more than one point. For $\tau \geq 0.07$ and $K \geq 8$, GM varies by at most 0.12. AIR and pointing accuracy keep improving in this range, but generation scores do not.

D.3 Paired Bootstrap over Prompts.

We test whether the gains of SST over LoRA FT exceed prompt sampling variation. For one training run per backbone, we resample the 800 GenEval2 prompts with replacement (B = 10,000) and compute paired per-prompt differences. All four intervals exclude zero (Table 14). The win rate is the fraction of prompts on which SST scores higher; no prompt is tied.

D.4 Human Evaluation

We validate Soft-TIFA and the improvement of SST with a blinded human evaluation on BLIP-3o 4B and BAGEL. For each backbone, we sample 80 GenEval2 prompts, 10 per atomicity level, with a fixed seed before inspecting any scores. Three non-author annotators judge each image of LoRA FT

Table 13: Sensitivity to the extraction hyperparameters on BLIP-3o 4B (single run each). AIR and Pt are measured against Grounded-SAM pseudo-references on the 1K Echo-4o subset of Table 12. AM and GM are GenEval2 scores after retraining with the resulting supports. Our setting is marked in gray.

	AIR	Pt	AM	GM
Temperature τ ($K = 8$)				
$\tau = 0.01$	0.341	0.643	56.28	14.87
$\tau = 0.07$	0.411	0.718	58.34	16.05
$\tau = 1$	0.433	0.726	58.26	16.01
Top-K ($\tau = 0.07$)				
$K = 1$	0.373	0.661	55.58	14.37
$K = 4$	0.403	0.693	57.62	15.82
$K = 8$	0.411	0.718	58.34	16.05
$K = 16$	0.432	0.732	58.38	16.13

Table 14: Prompt-level paired bootstrap of SST over LoRA FT on GenEval2 (800 prompts, $B = 10{,}000$, percentile intervals), for one training run per backbone. Win rate is the fraction of prompts on which SST scores higher.

Backbone	Metric	Δ (SST $-$ LoRA FT)	95% CI	Win rate
BLIP-3o 4B	AM	$+3.78$	$[+2.69, +4.86]$	59.4%
	GM	$+2.70$	$[+1.11, +4.28]$	60.9%
BAGEL	AM	$+1.04$	$[+0.27, +1.81]$	56.4%
	GM	$+0.82$	$[+0.11, +1.53]$	53.9%

and SST against the unchanged GenEval2 atom questions. Each image is presented independently, with model identity hidden and presentation order randomized. We take the majority vote per atom.

Soft-TIFA agrees with the human majority on 89% of the 2,448 atom questions (Cohen's κ = 0.76), on both LoRA FT and SST outputs. Human judgments favor SST at the atom level by 4.4 points on BLIP-3o 4B and 7.0 points on BAGEL, and both paired bootstrap intervals exclude zero (Table 15). The all-atom rate also increases on both backbones, but its intervals include zero at this sample size.

E Additional Qualitative Results

E.1 Phrase-Conditioned Support Mask Visualizations

Figure 6 illustrates the phrase-conditioned support masks extracted from the understanding path, complementing the schematic illustration in Figure 1. The masks are overlaid on the corresponding images, with brighter regions indicating stronger phrase support.

E.2 Failure Cases

We present a representative failure case at high atomicity to complement the quantitative trends reported in Section 5. Across all backbones, generation under dense compositional prompts at high atomicity remains challenging even with our framework. Figure 7 illustrates this regime through one such case.

F Limitations

SST inherits its limits from the components it relies on. It tunes LoRA parameters only, so its gains are bounded by the generative quality of the backbone. Its supervision comes from support masks that contain errors we cannot identify without phrase-level ground truth, and all reported results include these errors. Its scaffold modulates a continuous intermediate state, so purely autoregressive image-token UMMs would require a different design.

Table 15: Human evaluation of LoRA FT and SST on 80 GenEval2 prompts per backbone, 10 per atomicity level, sampled with a fixed seed before any scores were inspected. Human AM is the mean majority-voted satisfaction rate over atom questions. All-atom rate is the fraction of prompts on which the human majority marks every atom question as satisfied. Soft-TIFA scores are computed on the same subset and therefore differ from Table 1. On BAGEL, this subset shows a larger Soft-TIFA gap (+9.0 AM) than the full set (+1.04; Table 14). Agreement is the fraction of atom questions on which Soft-TIFA matches the human majority, computed separately on the images of each variant. Brackets are 95% prompt-level paired bootstrap intervals. Δ is computed before rounding.

Backbone	Metric	LoRA FT	SST	Δ [95% CI]
BLIP-3o 4B	AM (human)	58.3	62.6	$+4.4$ [$+1.5$, $+7.1$]
	AM (Soft-TIFA)	56.1	59.4	$+3.3$
	All-atom rate (human)	6.3	8.8	$+2.5$ [-1.3, $+5.8$]
	GM (Soft-TIFA)	9.5	11.5	$+2.0$
	Agreement	88.2	87.8	-0.5
BAGEL	AM (human)	70.0	76.9	$+7.0$ [$+2.7$, $+11.1$]
	AM (Soft-TIFA)	66.4	75.4	$+9.0$
	All-atom rate (human)	12.5	18.8	$+6.3$ [-1.3, $+14.2$]
	GM (Soft-TIFA)	20.2	28.8	$+8.6$
	Agreement	90.0	89.9	-0.2

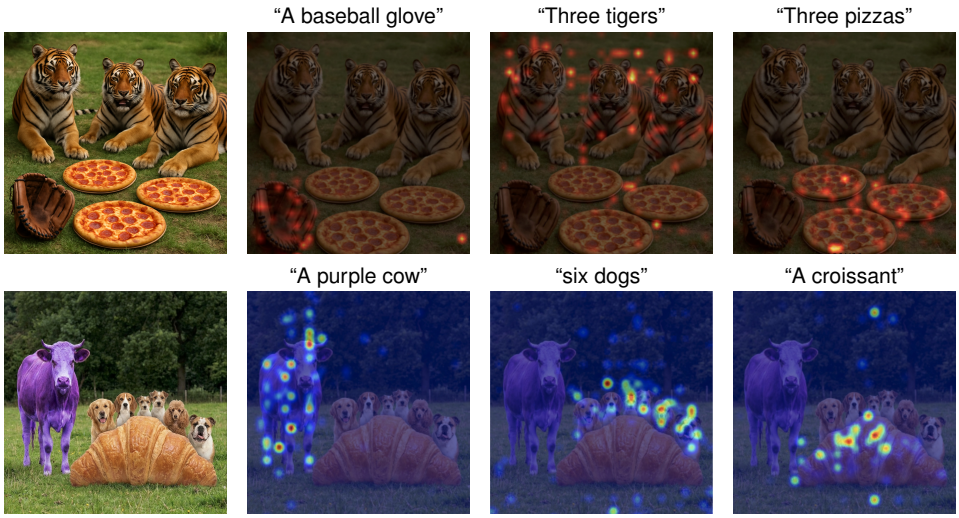


Figure 6: Phrase-conditioned support masks on images from the training distribution (Echo-4o; top) and the evaluation distribution (GenEval2 oracle image; bottom). The masks form spatially coherent regions across both distributions, supporting their use as pooling regions for phrase-local alignment.

Two questions remain open. Generation at high atomicity stays difficult on all backbones (Appendix E.2), which we attribute in part to the scarcity of dense prompts in current post-training data. Each training pair already includes an image that satisfies its caption, so denser prompts can be constructed and paired the same way. We also schedule the scaffold over training progress and do not analyze its contribution across diffusion timesteps.

G Broader Impact

Our work focuses on improving compositional fidelity in unified multimodal models, a foundational research direction with no immediate deployment path. Improvements in text-to-image generation more broadly carry well-documented risks, including the potential for misinformation, deepfakes, and copyright or privacy violations. We do not introduce new generative capabilities beyond those of the base UMMs we build upon, and we do not release new model weights; our framework only



Figure 7: Representative failure case at atomicity 8 on BAGEL. Prompt: **"three checkered cows, and two bicycles behind four pigs."** The base UMM and LoRA fine-tuning omit the "four pigs" phrase entirely, while SST generates pigs but with an incorrect count (three instead of four).

modifies the training procedure. Standard mitigations such as content provenance, watermarking, and access controls remain applicable to any downstream deployment of the base models.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state our main contributions, and the experimental sections support each claim.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are discussed in Appendix F.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The method is described in Section 4, with the extraction pipeline detailed in Appendix B, backbone-specific instantiations in Appendix A.2, and training configuration in Appendix A.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code will be released upon publication. The training data (Echo-4o) and evaluation benchmark (GenEval2) are publicly available, and we describe the experimental setup in detail.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: Training and evaluation details are provided in Section 5 and Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Tables 1 and 2 report means and sample standard deviations over three training runs; Appendix D.3 reports paired bootstrap intervals over prompts.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Compute resources are described in Appendix A.3.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader impacts are discussed in Appendix G.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release new pre-trained models or scraped datasets. Our framework modifies the training procedure of existing publicly available UMMs.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets, benchmarks, and pre-trained models used in this work are properly cited, and we use them in accordance with their respective licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not release new assets at submission time.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The human evaluation in Appendix D.4 asks three annotators to judge model outputs; it involves no crowdsourcing or research on human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Annotators judged model outputs; the study collects no information about the annotators and involves no risk beyond viewing ordinary generated images.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were not used as a core methodological component of this research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.